



## Preparando modelo y view

### Transcripción

[00:00] Hola a todos y todas, bienvenidos y bienvenidas a este nuevo video sobre servlet. En este caso, yo les había dicho que quería hacer un login dentro de nuestra aplicación, y bueno, la idea es tener algo parecido con lo que tiene Alura. Entonces por ejemplo, si nosotros vamos a [aluracursos.com/dashboard](https://aluracursos.com/dashboard) con esto, si ustedes están logados deberían llegar a la página principal donde ya tienen todos sus cursos y todo eso.

[00:28] Pero si yo ahora no estoy logado, por ejemplo, aparece directamente así, me aparece el login. Entonces, nosotros lo que queremos es eso, obtener un login, el cual nosotros no podamos acceder a los items, como listar empresas por ejemplo sin haber pasado antes por el login y haber iniciado sesión correctamente. Entonces vamos a hacer eso.

[00:56] Vamos a primero si vamos a trabajar con un login, nosotros necesitamos de usuarios, si vamos a necesitar de un usuario, entonces nosotros vamos a tener que crearlo acá en el modelo. Ya que tenemos una empresa, vamos a poder crear también un usuario. En esta creación del login vamos a pasar por todas las capas de nuestro modelo de nuestra arquitectura MVC.

[01:20] Entonces vamos a crear el usuario, yendo acá dentro de nuestro modelo, clic derecho new class, vamos a llamarlo de usuario. Finish y acá yo tengo ya un código copiado que les voy a dejar listo para que ustedes lo copien y lo peguen también en su proyecto.

[01:46] No es nada difícil, no es nada diferente de lo que ustedes hayan hecho ya, simplemente es al igual que la empresa tiene sus datos y getters y setters para obtener los datos, nosotros tenemos nuestros datos del usuario con login y contraseña, contraseña con N porque no podemos poner una Ñ acá en el código. [02:07] Podría tener muchos más datos, como cuál es la dirección de la casa, email. Bueno, varios datos, pero lo vamos a dejar simple acá, solo funcional para las cosas que realmente necesitamos para el login. Tiene sus getters y setters y tiene un método acá llamado esIgual, que lo que hace es comprobar si la contraseña del login y contraseña enviado realmente son iguales a los que pertenecen a ese usuario.

[02:35] No es nada más, simplemente estamos acá, haciendo, preguntando si el login es diferente de su login normal. Entonces ya devolvemos un false. Si la contraseña es diferente de la contraseña que él tiene devolvemos false y sin ninguno de esos ocurre entonces significa que todo está bien. Entonces devuelve un true, nada más, perfecto.

[02:55] Tenemos nuestro usuario. Ahora, ¿qué más necesitamos hacer? Tenemos que agregarlo a nuestra base de datos. Sería bueno iniciar ya nuestro servidor ya con una serie de usuarios y contraseñas listas. Entonces vamos a hacer eso, vamos a crear entonces esa lista de usuarios. Voy a copiar esta línea, "Ctrl + C", "Ctrl + V" desde listaEmpresas, va a ser listaUsuarios. Ahí está listaUsuarios, y es un arrays.

[03:29] Perfecto. ¿Que se está quejando acá? Vamos a dejarlo así y vamos a ver. Así como tenemos, estamos creando unas empresas y colocando datos de esa empresa, vamos a hacer lo mismo con nuestros usuarios, entonces, dentro de ese static vamos a colocar acá abajo usuario u1 = new Usuario(); faltó paréntesis, 1u.setLogin.

[04:08] El login va a ser mi nombre "bruno", ustedes pueden poner el nombre que ustedes quieran, u1.setContraseña y acá la contraseña va a ser ("12345"). Así como tengo el usuario 1, vamos a crear un usuario 2. U2, recuerden

cambiar acá el `u1` por `u1`. Vamos a colocar otro nombre acá pueden poner el nombre que ustedes quieran, yo voy a poner acá ("nayana"). Y `setContraseña("12345")`.

[04:41] Y ahora a esa lista de usuarios, `ListaUsuarios.add` vamos a agregar (`u1`); Y voy a copiar esto, "Ctrl + C", "Ctrl + V" (`u2`). Tenemos nuestras listas ya al comienzo con usuarios nuevos. Yo tendría que agregar acá un método, por ejemplo, para poder buscar un usuario y poder, por ejemplo, cambiar un usuario, pero eso lo vamos a hacer después.

[05:14] Por ahora solo quiero mantener esto bastante simple. Voy a guardar acá nuestro usuario, cerrar lista empresas acá y bien, tenemos entonces nuestro `modeloUsuario` y tenemos nuestra base de datos, un usuario, dos usuarios registrados en un comienzo. ¿Qué más tenemos que hacer?

[05:35] Bueno, lo que me gustaría hacer, por ejemplo, sería hacer la view, vamos a hacer la view. ¿Qué sería esa view? Sería un formulario para poder colocar un login. Entonces, vamos a ver dónde es que teníamos nuestras views. Acá, acá tenemos nuestras views. Tenemos nuestro formulario `NuevaEmpresa`, un formulario es un formulario a fin de cuentas, entonces simplemente vamos a copiar y pegar.

[06:07] `FormLogin` va a llamarse, `formLogin`. Vamos a abrir ese `formLogin` acá y veamos, primero tenemos nuestra entrada con link, ese action va a ser el action de la entrada. Eso está bien. Acá nosotros vamos a colocar un login y contraseña. Acá sí podemos poner la ñ. El nombre va a ser un tipo texto y el nombre va a ser login. Tienen que coincidir este campo recuerden con el del usuario.

[06:46] Y acá la contraseña va a ser del tipo "password". Esto nos va a permitir que en vez de que cuando nosotros escribamos se van a ver los puntitos de un password que siempre vemos en login, entonces con eso evitamos que se vea la contraseña, un motivo de seguridad. Y acá contraseña.

[07:07] Y la acción no va a ser nuevaEmpresa, la acción va a ser login. "Ctrl + S", perfecto. Tenemos nuestro login y ahora necesitamos que si nosotros hicimos nuestro model y nuestro view, ahora falta la parte del controlador, recordando que nuestro controller nosotros ya no vamos a tocar ese controller, nuestro servlet de entrada.

[07:28] ¿Por qué? Porque ya lo generalizamos para cualquier action, entonces lo que tenemos que hacer es crear un action. Entonces vamos a nuestra action, nuestra acción, entonces vamos a crear ese action: clic derecho, new vamos a ir a class, debería ser class, sí, new class. Vamos a crear el action login.

[07:54] Ese login ahora que nosotros tenemos una interfaz action, acción, podemos agregarla con el add. Entonces vamos a buscar acá acción. Ahí está, ahí tenemos nuestra interfaz, acción. Perfecto. Cando finalizamos, ya tenemos nuestro login que implementa la acción y ya tenemos nuestro método bonito de ejecutar.

[08:16] ¿Qué es lo que debería hacer esta acción de login? Bueno, ella va a recibir algunos parámetros. Vamos a tener el parámetro login y contraseña que recibimos de nuestro formulario, entonces vamos a colocar, vamos a hacer eso, `string paramLogin = request`. Esto ya es lo mismo que hemos hecho siempre, `request.getParameter`.

[08:45] Y acá vamos a colocar ("login"). Y así como creamos el parámetro login, vamos a también copiar esta línea, pegamos y vamos a tener el `paramContrasena`, y recibimos contraseña. Quiero ver si en el form puse contraseña, todo bien. Perfecto, tenemos los parámetros. ¿Ahora qué es lo que deberíamos hacer?

[09:17] Vamos a colocar acá solo un `system.out.println`, vamos a colocar un "login" y le vamos a agregar el nombre de la persona que se está logando, "paramLogin", perfecto, solo para para que se nos vea ahí que estamos logando. Perfecto. ¿Y ahora, qué es lo que deberíamos hacer?

[09:45] Lo que me gustaría a mí vamos a imaginar que el login fue okay, la persona, el usuario y la contraseña, el login y contraseña está correcto. ¿Qué es lo que debería ocurrir luego de eso? Luego de que está correcto, deberíamos tener una redirección hacia nuestra listaEmpresas. Vamos a colocar como que nuestra listaEmpresas es nuestra página principal.

[10:06] Entonces vamos a colocar el return, vamos a hacer un “redirect:entrada?acción=ListaEmpresas. Ese va a ser, vamos a hacer un redirect directo a listaEmpresas una vez que estuviera okay. Perfecto. Ya tenemos nuestro login, o por lo menos la lógica para poder loguear. ¿Pero qué es lo que ocurre?

[10:40] Nosotros cuando queremos mostrar un login nosotros necesitamos mostrar ese form. En ninguna de nuestras acciones estamos mostrando a nuestro formulario. Acá esta acción de login viene después de que la persona hace clic en entrar, en enviar el formulario. Entonces nosotros necesitamos una acción previa que en realidad muestre ese formulario de login.

[11:10] Y recuerden que es formLogin, nosotros no podemos acceder directamente desde una petición HTTP directamente a nuestro JSP. No podemos poner /gerenciador/formLogin. Nosotros tenemos que tener un action. Entonces vamos a hacer eso, vamos a crear una nueva acción, clic derecho, new class va a ser formLogin, sí, formLogin. Y también va a tener como interfaz action. Finish.

[11:45] Tenemos acá nuestro form de login. Acá solamente como la idea es solamente presentar, mostrar ese JSP, entonces vamos a hacer simplemente un forward. Vamos a colocar forward: y acá formLogin.jsp. En nuestro forward ya tenemos dentro de nuestro controlador servlet ya tenemos la lógica para poder encontrar ese FormLogin.jsp.

[12:20] Vamos a ver si está todo bien, vamos a probar esto. Bien, servidor, vamos a colocar un run acá, ya está todo guardado con "Ctrl + S". Vamos a

probar `gerenciador/` ¿Cuál sería nuestro login? `Entrada?accion=FormLogin`.  
Enter. A ver si funciona. Bien, tenemos nuestro `FormLogin`, vamos a inspeccionar el código: `entrada.jsession`.

[12:52] Tenemos un dato acá que no debería estar, vamos a ver sobre eso enseguida. Solo vamos a probar simplemente a ver si funciona. Login vamos a poner `bruno, 12345`, enviar, nos hizo una redirección. Google Chrome está diciendo que hay cosas que no deberíamos mostrar. Bien. Nos está redireccionando a `listaEmpresas`.

[13:18] Está yendo a la entrada `listaEmpresas`, si vamos a modificar está funcionando todo normal. Entonces nuestro form de iniciar sesión está funcionando. Vamos a ver, por ejemplo, si yo reinicio el servidor, quiero mostrarles una cosa que va a estar ocurriendo, porque no he hecho toda la implementación del login.

[13:42] Si yo me voy por ejemplo en una pestaña incógnita, vamos a usar ese `localhost gerenciador/entrada?accion=ListaEmpresas`, yo no debería poder entrar en `listaEmpresas`, ya que si nosotros vemos por ejemplo en Alura, `aluracursos/dashboard`, si nosotros vamos a `dashboard`, en realidad nos hace una redirección directo al login.

[14:05] No podemos entrar `dashboard` sin acceder a través de un login. Entonces, acá si yo me voy a la lista de empresas, vemos que sí estoy consiguiendo entrar en `listaEmpresas` y no estamos yendo a nuestro login, entonces esa es una cosa que vamos a tener que ver en el próximo video.